APPLICATION

FOR

UNITED STATES LETTERS PATENT

TITLE: CONTENT REQUEST ROUTING AND LOAD BALANCING FOR CONTENT DISTRIBUTION NETWORKS

APPLICANT: MICHAEL GILSON SLOCOMBE, VINCE FULLER, MATTHEW MILLER AND CASEY AJALAT

CERTIFICATE OF MAILING BY EXPRESS MAIL

Express Mail Label No. EL224699976US

I hereby certify under 37 CFR §1.10 that this correspondence is being deposited with the United States Postal Service as Express Mail Post Office to Addressee with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, Washington, D.C. 20231.

10-18-01
Date of Deposit

Leroy Jenkins
Signature

Leroy Jenkins
Typed or Printed Name of Person Signing Certificate

CONTENT REQUEST ROUTING AND LOAD BALANCING FOR CONTENT DISTRIBUTION NETWORKS

BACKGROUND

The invention relates generally to information retrieval in a network and, more particularly, to hosting and distributing content on a content delivery network such as the Internet.

The World Wide Web is the Internet's content retrieval system. In the Web environment, client systems effect transactions to Web servers using the Hypertext Transfer Protocol (HTTP), which provides clients with access to files (e.g., text, graphics, images, sound, video, etc.) using a standard page description language, for example, Hypertext Markup Language (HTML). A network path to a server is identified by a so-called Uniform Resource Locator (URL) having a special syntax for defining a network connection. Use of a Web browser at a client (end user) system involves specification of a link via the URL. In response, the client system makes a request to the server (sometimes referred to as a "Web site") identified in the link and, in return, receives content from the Web server. The Web server can return different content data object types, such as .gif and .jpeg files (graphics), .mpeg files (video), .wav files (audio) and the like.

As the Web server content provided by World Wide Web has continued to grow over time, so too has the number of users demanding access to such content. Unfortunately, the ever-increasing number of end users requesting Web content from Web sites has resulted in serious bandwidth and latency issues, which manifest themselves in delay to the end user.

To address these problems, many networking product and service providers have developed solutions that distribute Web site content across the network in some manner. One class of solutions involves replicating Web servers at multiple locations and directing traffic (by modifying the URL and forwarding, or using HTTP re-direct) to the "best" server based on a predefined selection policy, e.g., load balancing, network topology. Another class of solutions distributes content strategically and/or geographically, and often uses some type of centralized or hierarchical Domain Name System (DNS)-based site selection. The distributed sites include servers that perform reverse proxy with (or without) caching. One

such technique routes traffic to a content distribution site nearest the requestor by modifying URLs in the top-level Web page. Other DNS techniques use a round robin traffic distribution to distribute load to the content sites, but do not take into account the location of the requestor relative to those content sites.

SUMMARY

In one aspect, the invention provides a method of content delivery in a network. The method includes associating devices in a Domain Name System (DNS) with content server systems located in the network, the content server systems maintaining and serving content of a content provider, each DNS device configured to resolve the name of the content provider to an address for the content server system with which such DNS device is associated. The method further includes assigning to the DNS devices a common address, the common address being usable to resolve the name of the content provider such that a request for content of the content provider by a content requestor is sent to the content server system nearest the content requestor.

Particular implementations of the invention may provide one or more of the following advantages.

A performance benefit is gained because a content requestor can generally retrieve content from a content site closer than the origin server of the content provider. In addition, because there are multiple sites serving the content, the load from many end users is distributed among multiple systems and across different parts of the network. Also, an end user's DNS request can be routed to a content site nearest the requestor using pre-existing routing infrastructure. Because DNS uses a stateless protocol (UDP) for routing, the solution can handle anycast addressable caching without the problems associated with anycast service, namely, the protential packet-by packet load balancing site effects of protocols like TCP which maintain state information.

Other features and advantages of the invention will be apparent from the following detailed description and from the claims.

DESCRIPTION OF DRAWINGS

FIG. 1 is a depiction of a prior Web transaction.

FIG. 2 is a depiction of a prior Web transaction using reverse-proxy caching.

FIG. 3 is a block diagram of an exemplary content distribution network which includes content distribution nodes that support reverse-proxy caching and anycast address service to achieve distributed content delivery and load balancing.

FIG. 4 is a simplified block diagram of routing configuration software and associated table(s) (provided in the routing infrastructure of the content distribution network of FIG. 3) used to perform distributed content site selection based on anycast address routing and Border Gateway Protocol (BGP).

FIG. 5 is a depiction of an exemplary Web transaction using the content distribution network shown in FIG. 3.

Like reference numbers will be used to represent like elements.

DETAILED DESCRIPTION

The present invention features a content distribution mechanism for routing a content request to the nearest content distribution site in a content distribution network. It also provides for load sharing across multiple content distribution sites in a content distribution network. The content distribution mechanism has particular utility in and is therefore described within the context of an Internet-based, World Wide Web ("Web") network infrastructure.

Hereinafter, the following terminology is used:

"Content provider" refers to an entity having a Web site on a network. Generally, the entity produces the content for the Web site. The entity may operate a Web server system or may use the services of a hosting provider.

"End user" refers to a person who wishes to view the Web content. Typically, an end user uses a Web browser, such as Netscape Navigator or Internet Explorer, executing on a computer system, e.g., a personal computer.

"Domain Name System" (DNS) refers to a collection of systems available on the public Internet that can resolve a domain name to a specific Internet Protocol (IP) address, as is known in the art.

It will be understood that reference to distance on the network, such as one server being "closer" to an end user than another server, refers to a network distance. Thus, a

shorter distance implies a better path based on network criteria, and not necessarily a shorter geographic distance.

In the description to follow, a fictitious company "ABCD" is used as an example of a content provider having a Web site on a network.

FIG. 1 shows a conventional content delivery Web transaction 10. The transaction 10 involves a Web server 12 (also referred to as an "origin server"), an authoritative DNS system 14, an end-user system 16 and an end-user DNS system 18, all of which are coupled to a network 20, for example, the public Internet. The Web server 12 is a computer system containing content (e.g., Web pages) for a content provider, with the ability to provide this content in response to a Web request via the HTTP protocol. The authoritative DNS system 14 is a DNS system that can resolve domain names within the content provider's namespace.

For example, the DNS system 14 for the company "ABCD" would have information for host names ending in ".abcd.com" (such as "www.abcd.com"). Typically, the end-user system 16 is a computer being operated by an end user to perform Web "browsing" (that is, view Web pages). The end-user DNS system18 is a DNS server that the end-user system 16 uses to resolve domain names to IP addresses.

When the end user wishes to view a Web page or object (such as "www.abcd.com/PriceList"), the transaction 10 occurs as follows. First, the end user using the end user system 16 enters the name of a Web page into a browser (not shown) executing on the end user system 16. The end-user system 16 requests a DNS resolution for the host name ("www.abcd.com") from the end-user DNS system 18 ("DNS Req 22"). The end-user DNS system 18 determines which of the DNS systems that make up the DNS for the network 20 can resolve this host name by sending a DNS request to the authoritative DNS system 14 ("DNS Req 24"). The authoritative DNS system 14 resolves the name to an IP address and returns a response containing the IP address to the end-user 16 system via the end-user DNS system 18 ("DNS Resp 26"). The end-user DNS system 18, in turn, communicates the IP address to the end-user station 16 in a DNS response to the end-user system 16 ("DNS Resp 27"). The end-user system 16 contacts the Web server 12 at the specified IP address and requests the Web object ("www.abcd.com/PriceList") ("HTTP Req 28"). The Web server 12 returns the Web page corresponding to the requested Web object to the end-user system 16 ("HTTP Resp" 30). The browser running on the end-user station 16 displays the returned

Web page on the end-user system 16 for viewing by the end user.

FIG. 2 illustrates a Web transaction with reverse proxy caching 40. That is, the transaction 40 is the same basic Web transaction as shown in FIG. 1, but now employs a reverse proxy content server (shown as a cache server) 42 acting on behalf of the Web server 12. During reverse-proxy caching, the cache server 42 assumes the identity of the Web server 12 so that a Web request directed to the Web server 12 (in the example, "www.abcd.com") is instead directed to the cache server 42. This re-direction is accomplished by changing the entry for the Web site ("www.abcd.com") in the authoritative DNS system 14 so that the host name resolves to the address of the cache server 42 instead of the address of the original Web server 12. A new name is assigned to the address of the original Web server 12 (e.g., "origin.abcd.com"). Thus, 22, 24, 26 and 27 are the same as in FIG. 1. However, the IP address returned by the authoritative DNS system 14 is that of the cache server 42, not the Web server itself, as was previously described with reference to FIG. 1. Consequently, instead of sending the subsequent content request to and receiving a response from the Web server 12 (as was shown in steps 28 and 30 of FIG.1), the end-user system 16 sends the content request to the cache server 22 ("HTTP Req 44") and the cache server 22 fetches the requested content from the Web server ("HTTP Req 46"). The Web server 12 returns the requested content to the cache server 42 ("HTTP Resp 48"). The cache server 42 completes the transaction by serving the requested content to the end-user system 16 ("HTTP Resp 50") and caching the content so the content will be readily available to support future requests for the same content ("store"52).

FIG. 3 shows a content distribution network ("CDN") 60. The CDN 60 includes at least one Web server, shown as a Web server 62, a DNS system 64 for Web server 62, and end-user stations 66a and 66b, all connected to a network 72. The network 72 is implemented as the public Internet. The end-user stations execute Web browsers 68a and 68b, respectively. Each end-user station 66 has an associated end user DNS, however, only an end user DNS 70a for the end-user station 66a is shown. Of course, additional end-user stations may be connected to the network 72. Also connected to the network 72 are multiple content distribution nodes 76a, 76b and 76c, which support distributed content delivery services for one or more Web sites on the network 72. The CDN nodes 76 interact with the origin server 62 containing the original Web content, the various DNS systems 64, 70 and, of

course, the end-user systems 66.

Each of the CDN nodes 76 includes a DNS system 78 coupled to and associated with a Web content server system or site 80. In one embodiment, as described herein, each content server system 80 is implemented as a cache server system. The techniques described herein could also apply to other types of content servers, such as mirrored Web content servers, or Web content servers having different content (e.g., customized for geographic area). Each DNS system 78 in each node holds a table that includes an address entry which the DNS system 78 uses to map the domain name of the content provider to the IP address of the cache server in that same node. Although only one such Web site (Web site 62) is shown, it will be appreciated that other Web sites may be connected to the network 72 and use the DNS and content caching services of the nodes 76, as will be described. The nodes 76 are deployed at different locations in the network 72. Preferably, the nodes 76 are geographically distributed across the network 72.

Optionally, the CDN 60 may include a CDN manager 82 that can be used by a network administrator (for example, a CDN node hardware and/or CDN node service provider) to configure the CDN to use the CDN nodes.

The network 72 is intended to represent a simplified view of the Internet. In the simplified depiction of FIG. 3, the network 72 includes a plurality of interconnected routers or routing networks, e.g., routers 74a. 74b, ...,74g, for routing packets from one domain to another within network 60. In actuality, the Internet is made up of many private "routing networks" (networks including one or more routers, and possibly other types of networking components as well), e.g., local, regional and centralized Internet Service Providers (ISPs), some of which are connected to Network Access Points (NAPs) or exchanges, or each other at public or private peering points. In the simplified Internet configuration shown in FIG. 3, routers 74a, 74b, 74c 74d and 74e are located at network entry points. The end user station 66a and end user DNS system 70a connect to the network 72 via the router 74a. The end user station 66b is coupled to another router 74i, which connects to the router 74d. The Web Server 62 and associated DNS system 64 are connected to the network 72 via the router 74b. Preferably, to the extent possible, and for reasons which will be discussed below, the end user DNS systems such as system 70a are located near the end user systems with which they are associated.

Also, preferably, the geographically dispersed nodes 76 are located so as to be as close as possible to various network entry points, exchanges or both. The network entry points each may correspond to an ISP Point of Presence (POP). In FIG. 3, for illustrative purposes, the nodes 76a, 76b and 76c are shown as being connected to entry point routers 74c, 74b and 74d, respectively, but need not be directly connected to network access routers in this manner.

The caching servers 80 have unique IP addresses. The DNS systems 78 share a common IP address as well as have unique IP addresses. The end-user DNS systems, e.g., end user DNS system 70a, resolve to the common address. That is, the end-user DNS system 70a knows which DNS system (in this example, the DNS system 64) has an address for a high level domain server, e.g., .com, .org, and maintains tables of all domain names and knows which server (authoritative DNS server) to consult for the address of the domain server. Thus, the address lookup table in the DNS system 64 is configured to indicate that a server corresponding to the common address can resolve the domain name of the content provider to an IP address.

One way to implement this content distribution configuration is to use an anycast address as the common address. An anycast address is a unicast address that is used in multiple places. Various Internet Engineering Task Force (IETF) Internet Requests for Comments (RFCs) describe implementations of anycast addresses in IP networks. The IETF is a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet. The following anycast-related RFCs are hereby incorporated by reference in their entirety for all purposes: RFC 1546 (November 1993); RFC 2372 (July 1998); RFC 2373 (August 1998); and RFC 2526 (March 1999).

As described in RFC 1546, an anycast address may include a subnet prefix identifier and an anycast identifier. The subnet prefix may be used to specify the network providing the anycast addresses. The anycast identifier is used to specify one of many possible anycast addresses on a particular subnet. A unicast address, or conventional IP address, specifies a single interface on a computer network. In contrast, an anycast address may specify more than one interface. For example, anycast addresses may be used to specify a group of one or more servers on a computer network. These servers may provide a redundant service.

Routers forward packets destined to anycast addresses to the closest anycast destination for a particular address. Thus, anycast addresses provide a way to distribute load across one or more servers

The anycast address is advertised to the network 72 from each node 76 using a dynamic routing protocol, the Border Gateway Protocol (BGP). The BGP is a routing protocol used to exchange network reachability information between Internet border routers. It enables those routers to make intelligent routing decisions as to the best path. The BGP is used by such routers as their exterior routing protocol in order to advertise routes to other border routers. BGP uses TCP as its transport protocol for exchanging routing information. Adjacent routers running BGP set up a TCP connection to exchange entire routing tables. Each router has enough information to determine an optimal next hop to a destination. The BGP is also described in various RFCs, including RFC 1267 (October 1991) and RFC 1654 (July 1994), incorporated herein by reference.

Referring to FIG. 4, routing configuration support 90 in a router such as router 74b includes a routing algorithm (software) 94 and a routing table 92. The router 74b receives address path information from the nodes 76 in accordance with the BGP route information exchanges and updates 96, including the anycast address 98 and associated paths 100 for each of the nodes, and stores them in the routing table 92. As a result of the BGP table information exchanges and updates, routers in the network 72 maintain pointers (that is, the paths or routes) that allow it to determine the next hop to every unique address in the network, as well as multiple pointers to the anycast address. The routing algorithm 94, in response to receipt of a DNS request packet 102 from an end-user system 66 for resolution of a DNS name for the content provider, uses the path information 100 stored in the routing table 92 to a select a path to the nearest CDN node. Thus, a router may see multiple connections to the anycast address, but selects the path that represents the shortest network distance (e.g, the topologically shortest path). More specifically, as this routing occurs as part of a DNS resolution, the router selects a route to the closest DNS system78. Because the selected DNS system resolves to the address of the cache server in the DNS system's node, the DNS anycast routing, in essence, serves to select the local content site (cache) from which content will be served.

Referring now to FIG. 5, a Web transaction 90 occurring over a CDN (such as the

CDN 60 shown in FIG. 3) is illustrated for the running example of the "ABCD" Web site. To provide content distribution service for this site, the following configurations are implemented. The original Web server 62 is renamed in the DNS system 64 to reflect a change in its role from being the primary Web server to becoming the original content source for the cache servers 80. The domain "www.abcd.com" is renamed "origin.abcd.com". In the end-user DNS system 70a, an entry is made to indicate that the CDN DNS system 78 is the authoritative server for the end user web domain. For example, in the dns.abcd.com system, the entry for "www.abcd.com" no longer resolves to a specific address but instead refers the content requestor to the CDN DNS server 78 at the anycast address of 4.2.2.19. The CDN DNS server entry resolves to the address of the associated node's caching server 80. As a result, each node 76 resolves the Web site name to a different address. In the above example, DNS server 78a in node 76a (node 1) resolves the name "www.abcd.com" to 10.3.15.1, whereas in node 76b (node 2), the DNS server 78b resolves the same address to 10.3.15.2. In this manner, therefore, the entry made in the CDN DNS server 78 in each node enables distribution of the Web site among the cache servers 80 in their respective nodes and locations, as will be described in further detail below.

Once the appropriate configurations have been completed, it is assumed that the Web site www.abcd.com is being handled by the CDN nodes 76. Each node 76 advertises the anycast address of its own DNS server 78 to the network 72 using the BPG protocol, as discussed above. The address of the DNS server at each node is identical. That is, from a network point of view, the network "thinks" it is connected to a single host at multiple points.

Referring to FIGS. 3 and 5, the Web transaction operation 90 is as follows. When an end user of the end-user station 66a requests an object from the now "accelerated" Web site, the end-user system 66a first resolves the content provider name via DNS. That is, the end-user station (acting as a content requestor) sends a request to the local DNS server 70a ("DNS Req 92"). That server 70a resolves the name, ultimately by sending a request to the CDN DNS anycast address. The DNS server 70a sends a request to the ABCD DNS server 64 ("DNS Req 94"), which returns the anycast address of the CDN nodes 76 to the requesting DNS server 70a (DNS Resp 96). The DNS server 70a then transmits a DNS name resolution request addressed to the anycast address and that request is generally routed (by various ones of the network routers 74a – 74h) to the CDN node nearest the user's DNS

system ("DNS Req 96"). In this instance, because the request enters the network via the router 74b, the router 74b determines that the shortest path to the anycast address is the path to the node 76b (node 2). At that node, the DNS server 78b resolves the name to the cache server address for that node, that is, IP address 10.3.15.2 assigned to the cache server 80b, and returns the cache server address to the end-user DNS server 70a ("DNS Resp 98"). The end-user DNS server 70a forwards the address to the end-user system 66a ("DNS Resp 100). From this point on, the remaining steps of the transaction are much the same as steps 44 through 52, shown in FIG. 2. That is, the end-user system 70a requests the Web object from the cache server in the nearby node where the DNS name was resolved, i.e., node 76b, and that node's cache server 80b, in turn, checks for a cached version of the object (if cached content already resides on the cache server). If no content has yet been cached in the server, or the cached copy is stale or otherwise invalid, the cache server 80b retrieves the Web object from the origin server 62, and serves the object to the end-user system that requested it.

Typically, as is known in the art, Web content can be marked with certain caching attributes, for example, whether or not the content is at all cacheable, how long the content may be held in cache. In the case of the former attribute, if the content is marked as uncacheable (e.g., dynamic or content containing sensitive information), the cache server discards the content after serving it to the requestor. Otherwise, if the content is cacheable, the cache server will store the content in local storage and maintain the cached content according to any other cache attributes. Content can be localized, for example, using ad insertions with local content. Content in the caches can be pre-loaded (all the cache servers receiving the same content). That is, the content can be replicated on all cache servers so that even the first request will have a fast response time. Preferably, the caches are not pre-loaded with content but instead build their cached content based on user requests/usage over time. Content is retrieved from the origin server 12 when a first user request is received, and then stored locally. If subsequent requests are received for the same content, the cached copy is used if it is still valid, as was mentioned earlier. Thus, the cache server need not retrieve the content from the origin server again. Each cache server contains a translation table so that it knows where to retrieve any particular Web page from the origin server. For example, the cache server 80b would know that the page "www.abcd.com/PriceList" can be retrieved from "origin.abcd.com/PriceList".

Preferably, the CDN node contains software to monitor the load in various parts of the node cache system (disk, CPU, I/O, et cetera) by determining at least one load metric value (based on metrics such as utilization, latency, etc.) and comparing each such metric value to a predefined overload threshold. Upon reaching a predefined overload threshold, the monitoring software informs the routing software in the CDN DNS server to withdraw its BGP routing advertisement.

Thus, under normal conditions, all CDN nodes are advertising the address of their DNS servers to the network, and so a DNS request will be directed to the nearest CDN node. If a node becomes heavily loaded and detects an overload condition through its internal monitoring, the node stops advertising its DNS address to the network so that no further requests will be directed to that node. Consequently, DNS requests that normally would have been routed to that node as a first choice are routed to the next closest active node.

This overload detection and load balancing mechanism has the advantage that Web transactions already in progress are not interrupted by a shift in resources. Any system that has already resolved a DNS name to the now inactive node will continue using that node until the DNS name expires. The load in that node will slowly decrease until such time as the node can start accepting new clients, at which time it will start advertising its DNS system address to the network again.

Other embodiments are contemplated. For example, it is possible to use an anycast scheme with the cache servers themselves. With reference to the system shown in FIG. 3, the network routing directs the DNS request from the end user's DNS system, such as system 70a, to the closest of the CDN nodes 76. Although the node is the node closest to the end user's DNS system, it may not necessarily be the closest to the end-user system. In those cases where the end user's DNS system is a substantial distance from the end user, it is possible that the end user system will use a CDN node that is not the closest one to the end user system. Allowing the cache servers to use a common address would ensure that the end user's Web request is indeed routed to the nearest CDN node. There is a significant drawback associated with using the anycast-addressable cache server approach, however. The client/server portion of the transaction uses the TCP protocol, thus requiring multiple exchanges between the end user and the cache server to complete a transaction. With anycasting, there is no guarantee that subsequent packets in a transaction will be routed to the

same server. In cases where the packets are split between two or more cache servers, a successful transaction cannot occur. In contrast, although requiring that the end user DNS system be located in close proximity to the end user system for optimal CDN performance, the anycast-based DNS resolution is completed using a single packet exchange with the stateless UDP protocol, thus eliminating the packet-by-packet load distribution problem seen with TCP.

In addition, the Web caches, i.e., the cache systems 80 each can be implemented to include multiple caches servers connected, for example, in a cluster configuration. There may be multiple servers available to support one customer (origin server) or, alternatively, one or more cache servers available to support multiple customers' content cached at one node (site). In yet another alternative, the cache server clusters can include a switch to select from among the cache servers in a given node/cluster based on a predetermined selection policy, for example, content-aware selection (which enables the clustered servers store different content, and maps requested objects to the appropriate servers), load balancing, and so forth, using known techniques.

Other embodiments are within the scope of the following claims.

What is claimed is: